International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)
ISSN(P): 2250-1568; ISSN(E): 2278-9448
Vol. 3, Issue 5, Dec 2013, 31-40

OPTIMIZATION OF OPTICAL CHARACTER RECOGNITION FOR PRINTED DEVANAGARI TEXT USING ANFIS TECHNIQUES

GANESH S. SABLE[1] & SHEETAL ARUN NIRVE[2]

[1]Department of Electronics Engineering, Savitribai Phule Women's Engineering College, Sharanapur, Maharashtra, India

[2]Department of ETC, Deogiri Institute of Engineering and Management Studies, Aurangabad, Maharashtra, India

ABSTRACT

Character recognition is quite difficult task. It is the process of distinguishing the input character as per their predefine character class. There are different researchers who work on English language in last few years. Which results in technology whose practical application is possible. But if we will talk about devnagari script it is seen that due to its complicated structure very less work is there. Devnagari is the third most language spoken in word, therefore this system is proposed to develop tech. which will give maximum accuracy in minimum time period with less cost. This system could be use in practical life for recognition of printed information present on documents like cheques, envelopes, forms, and other manuscripts has a variety of practical and commercial applications in banks, post offices, libraries, and publishing houses. Here we are using multiple feature instead of utilizing single feature we are using various features like GLCM, color dominant, Affine movement invariant and Histogram.

KEYWORDS: Character Recognition, Printed Devnagari Text, ANFIS, GLCM, Affine Moment Invariant, Histogram

INTRODUCTION

Character recognition is the process to classify the input character according to the predefine character class. With increasing the interest of computer applications, modern society needs the input text into computer readable form. This research is a simple approach to implement that dream as the initial step to convert the input text into computer readable form. Some research for hand written characters are already done by researchers with artificial neural networks. Rapidly growing computational power may enable the implementation of CR methodologies. Digital document processing is gaining popularity for application to office and library automation, bank and postal services, publishing houses and communication technology. English Character Recognition (CR) has been extensively studied in the last half century and progressed to a level, sufficient to produce technology driven applications. But same is not the case for Indian languages which are complicated in terms of structure and computations. Devanagari being the national language of India, spoken by more than 500 million people, should be given special attention so that document retrieval and analysis of rich ancient and modern Indian literature can be effectively done.

SYSTEM DEVELOPMENT

Due to various font size, writing style and similar shape of characters it is difficult to recognize the character. There for here this system is proposed to develop which will give maximum accuracy with in minimum time period. Which applications in the data present on paper documents has to be transferred into machine-readable format. Automatic recognition of printed and handwritten information present on documents like cheques, envelopes, forms, and other manuscripts has a variety of practical and commercial applications in banks, post offices, libraries, and publishing houses.

Here we are using multiple feature instead of utilizing single feature we are using various features like GLCM, color dominant, Affine movement invariant and Histogram.

Proposed Character Recognition System

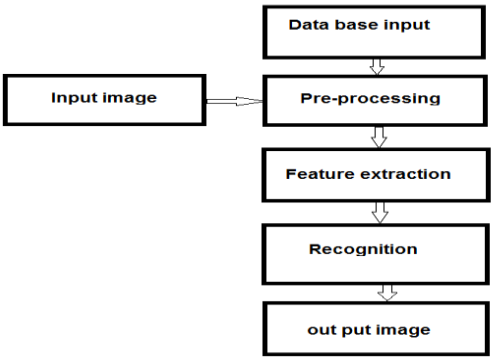


Figure 1: Block Diagram of Proposed Character Recognition System

Block diagram given above represents the working flow of proposed system. In first block we will select image as input image for feature extraction to generate database. In next block preprocessing will be there on selected image i.e. conversion of color image to binary image, then filtering of converted image. Then next block consist of extraction of features like GLCM, Affine moment, Histogram. After extracting features they are stored in .mat file. Now again new input image is selected for character recognition purpose again preprocessing steps are repeated. Finally in recognition block characters are recognized, and recognized character is shown in last block i.e. o/p image.

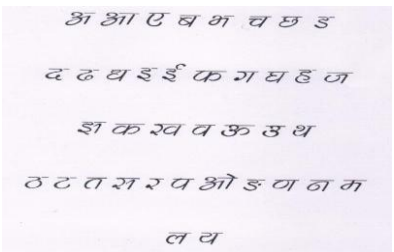
PERFORMANCE ANALYSIS

Introduction

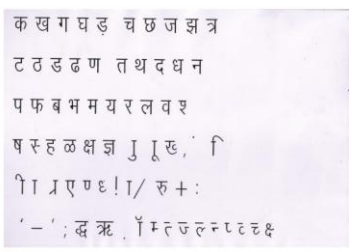
The aim of this character recognition system is to develop the character recognition algorithm that achieves the high accuracy & high recognition speed. Also the graph of feature extraction, neural network training and recognition rate of character.

Data Base

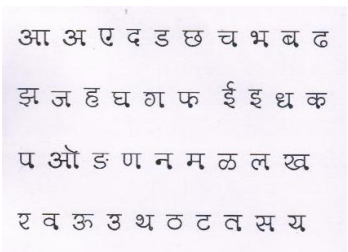
I have selected characters of different font size and font face for generating database. The following figure shows the data base used. I have installed the kruti devnagri software in my system. Then characters are typed in word document and snap shot is taken. After cropping the required part image is saved by name new database. Figure 2(a) (b) (c) (d) (e) shows the image used to generate database.



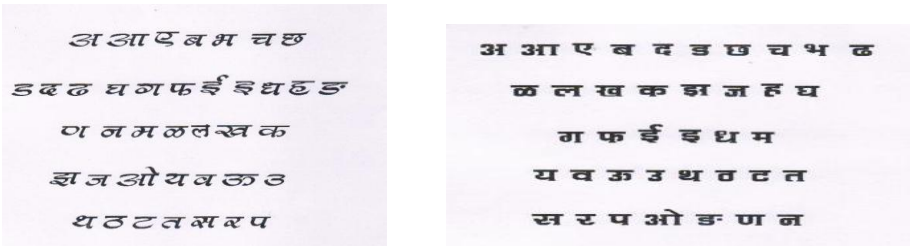
(a)



(b)



(c)



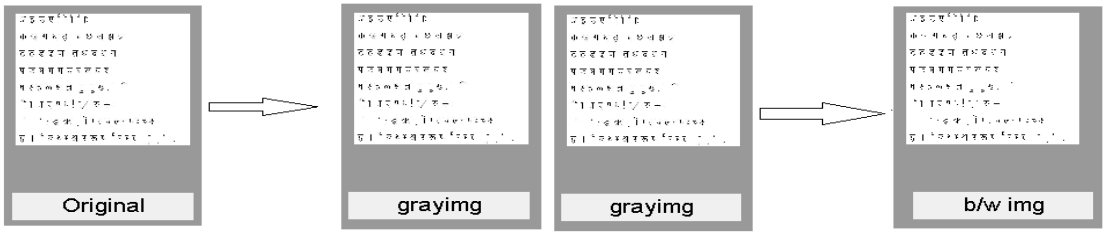
(d) (e)

Figure 2: (a) (b) (c) (d) (e) Database Images

Pre-Processing

Converting Original Image to Gray Scale Image to Binary Image

As shown in the following figure 3 (a), (b) (c) represents the pre-processing step. First of all the complete set of character is converted into gray image.



(a) (b) (c)

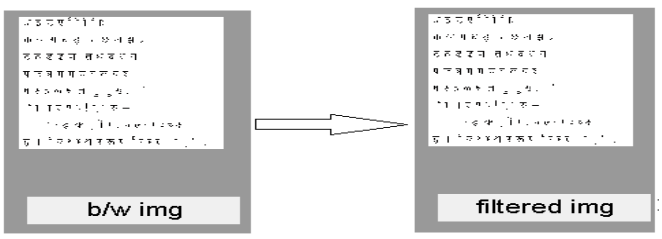
Figure 3: (a) Original Image (b) Gray scale Image (c) Binary Image

Converting Gray Scale Image to Binary Image

Then second step is converting gray image to binary image for that, MATLAB code is used which generates a Gray-decoded output vector or matrix y with the same dimensions as its input parameter x. After binarization image is converted in to 0's and 1's format.

Filtering of Binary Image

Now next step is removal Figure 4 (a) represent the noise free image. Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges.



(a) (b)

Figure 4: (a) Binary Image to (b) Filtered Image

Now it is not possible to show parameter of each character in report there for only few character are shown in diagram. Now each line is separated by using the line by line segmentation, from complete figure as shown in the following figure. Then each character is separately segmented by character by character segmentation process, from single line. And simultaneously, centroid of each character is find out which is denoted by red cross and box around the character is represented by blue colour.

Figure 5: Separation of Each Character Using Character by Character Segmentation

Following MATLAB windows represents the samples taken to generate data base. The value of each character is stored in structure files and denoted by <1×1struct>. The structure consist of Area occupied by character, Centroid, Bounding Box, Eccentricity, Orientation, Binary values of image, and perimeter. Each character have unique value of this perimeters. All this values are stored in findchardata file. Format of findchardata file is .mat file, Shown in figure 6.

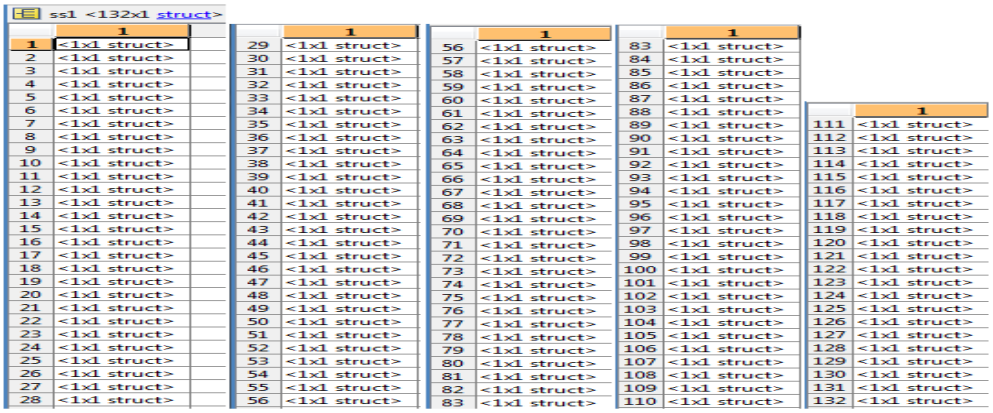


Figure 6: Sample Taken to Generate Database

Post-Processing

Feature Extraction of Character

Each <1×1> struct present in the table consist of values of co-efficient of binary image, Area, centroid, eccentricity, orientation, bounding box, perimeter of single character. It is not possible to represent feature of all character in this report therefore we will represent two characters with their details of parameters. If we will go step by step the first parameter for character प is area i.e. 187. Then come to second step the centroid of character is given by <1×2 double>.

Table 1: Parameters of Character प

ss1(43,1) <1x1 struct>

Field	Value	Min	Max
Area	187	187	187
Centroid	[34.9412,89.6417]	34.9412	89.6417
BoundingBox	[24.5000,74.5000,17,34]	17	74.5000
Eccentricity	0.8014	0.8014	0.8014
Orientation	-67.3618	-67.36...	-67.36...
Image	<34x17 logical>		
Perimeter	132.0416	132.04...	132.04...

Table 2: Centroid of Character प

ss1(43,1).Centroid <1x2 double>

	1	2
1	34.9412	89.6417

The bounding box of each character gives the boundary of character is given by <1×4 double>. The value of eccentricity is 0.8014 for this particular character. Orientation of this character is -63.3618. The value of perimeter is 132.0416.

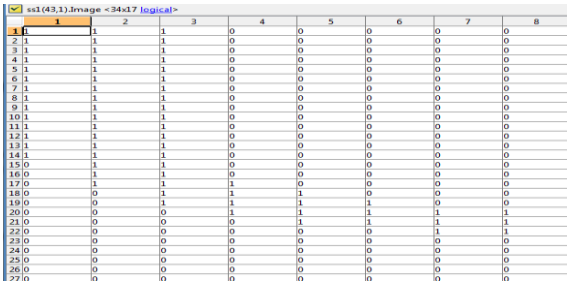
Table 3: Value of Bounding Box प

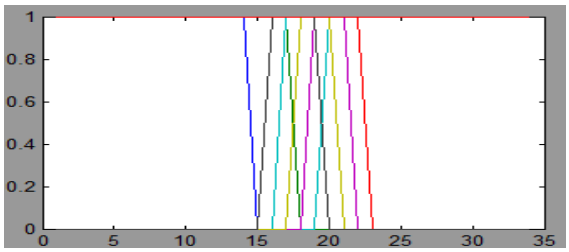
ss1(43,1).BoundingBox <1x4 double>

	1	2	3	4
1	24.5000	74.5000	17	34

Following table shows, the binary values of character प it is < 37×17 logical > image, the graph plotted gives the graphical representation of this character.

Table 4: Binary Value of Character प





Graph 1: Binary Value of Character प

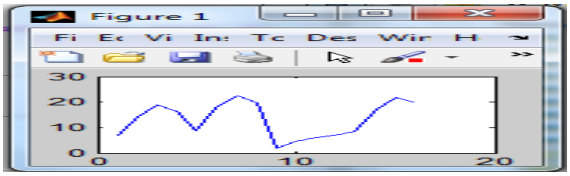
GLCM of Character

Now the following graph shows the GLCM. In GLCM the present texture of character is texture correlation as function of offset. The gray level co-occurrence matrix represents the values of 'contrast', 'correlation', 'energy', 'homogeneity' of each character which is stored in datafile.mat file. We took example of two characters there for, features of only two characters are shown here.

Table 5: Value of Contrast of प

stats.Contrast <1x2 double>

	1	2
1	4.5776	14.1278



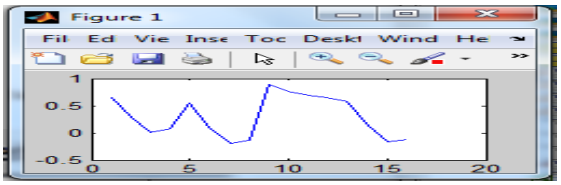
Graph 2: Graph of Contrast of प

Table 6: Value of Correlation of प

Variable Editor - CRL

CRL <1x16 double>

	1	2	3
1	0.6468	0.2945	0.0071
2			

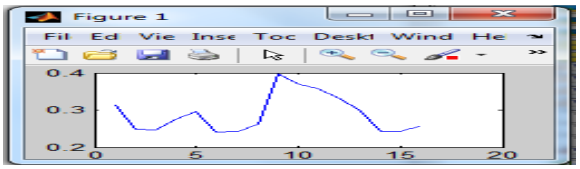


Graph 3: Graph of Correlation of प

Table 7: Value of Energy of प

E <1x16 double>

	1	2	3
1	0.3128	0.2491	0.2458
2			

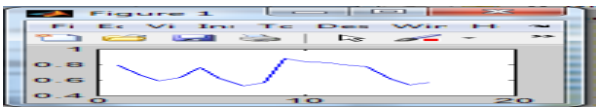


Graph 3: Graph of Energy of प

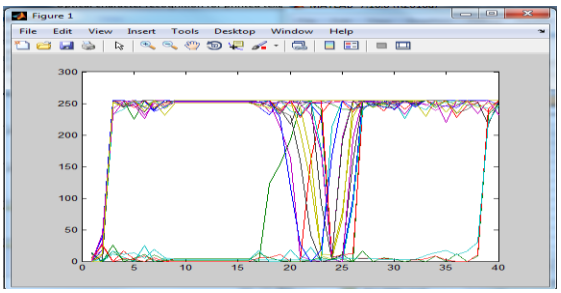
Table 8: Value of Homogeneity of प

stats.Homogeneity <1x2 double>

	1	2
1	0.8442	0.6751



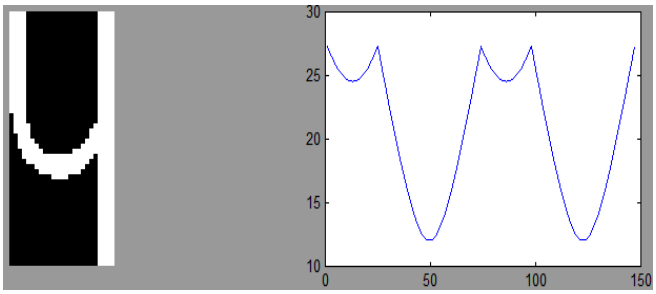
Graph 4: Graph of Homogeneity of प



Graph 5: Extracted Feature of Character प

Histogram of Character

Now it is not possible to show histogram of each character in report there for only few character are shown in diagram as follows:



Graph 6: Histogram of प

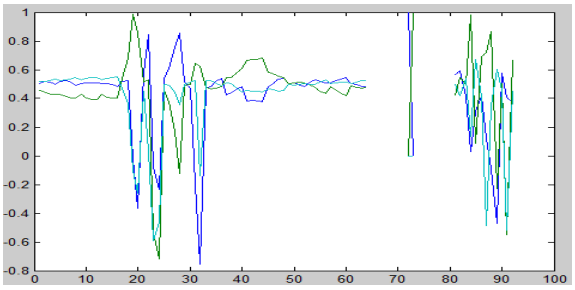
Input to Neural Network

Feature extracted from the character are provided to the input of neural network. Here ANFIS code is used, in which membership function is used, and no. of epoch's are decided. As shown in below table input value of neural network is in digit form and graphical representation of the numeric value is also shown below.

Table 9: Input Values Provided to NN

inputs <92x4 double>

	1	2	3	4
1	0.5052	0.4528	0.5195	0.5195
2	0.5159	0.4447	0.5178	0.5178
3	0.5139	0.4311	0.5244	0.5244
4	0.4977	0.4265	0.5340	0.5340
5	0.5180	0.4300	0.5229	0.5229
6	0.5232	0.4139	0.5268	0.5268
7	0.5128	0.4036	0.5358	0.5358
8	0.4934	0.4037	0.5448	0.5448
9	0.5037	0.4270	0.5310	0.5310
10	0.5092	0.4005	0.5387	0.5387
11	0.5095	0.3905	0.5422	0.5422
12	0.5053	0.3924	0.5435	0.5435
13	0.5017	0.4288	0.5312	0.5312
14	0.5014	0.4098	0.5388	0.5388
15	0.4954	0.4011	0.5448	0.5448
16	0.4795	0.4008	0.5520	0.5520
17	0.5181	0.5409	0.4685	0.4685
18	0.5235	0.6822	0.3609	0.3609
19	-0.0194	0.9889	-0.1042	-0.1042
20	-0.3666	0.8551	-0.2593	-0.2593
21	0.5756	0.5134	0.4500	0.4500
22	0.8461	0.5307	0.0355	0.0355
23	-0.0818	-0.5352	-0.5945	-0.5945
24	-0.2355	-0.7207	-0.4610	-0.4610
25	0.5401	0.4615	0.4976	0.4976
26	0.6102	0.3775	0.4925	0.4925
27	0.7568	0.1705	0.4462	0.4462



Graph 7: Input Values Provided to NN

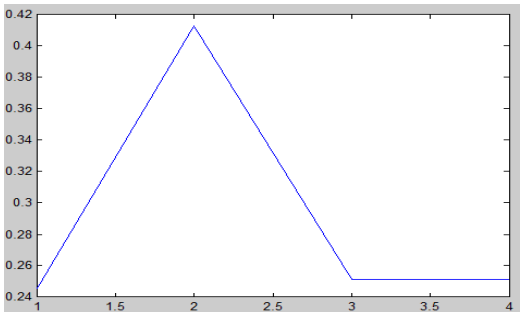
Output of NN

Output of neural network can be represented by following graph. As well as recognized output is also shown in following figure (a), (b).

Table 10: Output of NN

outputs <1x4 double>

	1	2	3	4
1	-0.0577	0.2819	-0.3532	-0.3532



Graph 8: Output of NN

Figure 7: Final Output i.e. 100% Recognised Character

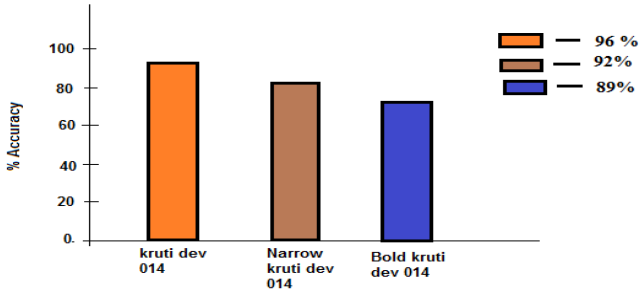
Recognition Accuracy (Recognition Rate)

Recognition accuracy = (No. of correctly recognized speech samples/Total no. of testing samples) x100. For the best recognition system, Recognition accuracy should be high. It is calculated as ratio of number of correctly recognized speech samples upon total number of samples used in testing.

Table 11: Character Recognition Test Results

Sr. No	Character	Kruti dev 014	Narrow kruti dev014	Bold kruti dev	Time Required in sec.	Sr. No	Character	Kruti dev 014	Narrow kruti dev014	Bold kruti dev	Time Required in sec.
1	अ	T	T	T	4.2	15	ख	T	T	T	5.1
2	आ	T	T	T	4.5	16	ग	T	T	T	5.4
3	इ	T	T	T	3	17	घ	T	T	T	4.8
4	ई	T	F	T	3.8	18	ढ	T	T	T	4.3
5	उ	T	T	T	5	19	ढ	T	T	T	5.1
6	ऊ	T	T	T	5.3	20	च	T	T	T	4.7
7	ए	T	T	T	4.4	21	र	T	T	T	3.8
8	ऐ	T	T	T	4.9	22	ज	T	T	T	3.6
9	ओ	T	T	T	6	23	त	T	T	T	4.1
10	अं	F	F	F	5.2	24	ह	T	T	T	5
11	अः	T	T	F	5	25	ठ	F	F	F	5.8
12	ऋ	T	T	T	6.3	26	ल	T	T	T	5.3
13	श	T	T	T	4	27	न	T	T	T	4
14	क	T	T	T	4.3	28	प	T	T	T	5.6

Character Recognition Accuracy Graph



Graph 9: Character Recognition Accuracy Graph

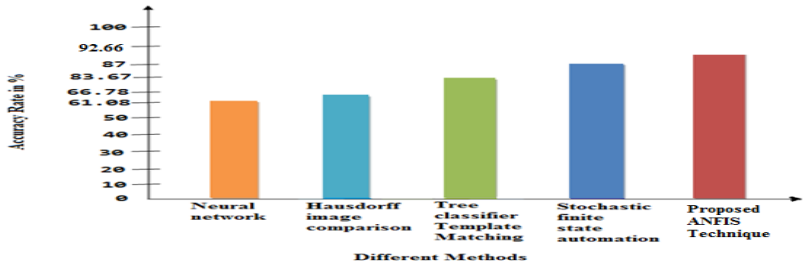
Comparison with Existing Systems

The accuracy of proposed system is compared with the accuracy of existing system.

Table 12: Comparison with Existing Systems

Reference No.	Feature	Classifier	Accuracy (%)
[14]	GSC	Neural Network	Recognition rate was 61.8%.
[15]	STRUCTURAL AND STATISTICAL	Hausdorff image comparison	Recognition rate was 66.78%
[12]	STATISTICAL	Tree classifier and template matching	Recognition rate was 83.67%
[14]	SFSA	Stochastic finite state automation	Recognition accuracy was 87%
Proposed system	STRUCTURAL AND STATISTICAL	Artificial Neuro Fuzzy Interfacing System	Recognition accuracy Average is 92.66%

Comparison of Character Recognition Accuracy among Different existing Methods:



Graph 10: Graph Representing Comparison between Different Existing Methods

The above graph 4.19 representing the different accuracy results of different methods. From this it is seen that proposed method gets the high accuracy 92.66% as compared to existing method.

CONCLUSIONS

The Character recognition is one of the difficult tasks, because verity font size and font faces are present now a day. So it's a try to achieve maximum accuracy and reduce time duration required in recognition of character. The proposed method hopefully can inspire a new thinking and new way to tackle the face recognition problem. The performance of the proposed method in terms of recognition accuracy is obtained. Features used in character recognition i.e. GLCM, Colour dominant, Histogram, AFFINE moment invariant, gives good results compare to others, and for recognition process ANFIS (Artificial neuro fuzzy interference system) tech. is used which gives the best result compare to other technique.

Talking about only two characters i.e. **प** and **न** it gives approximately 92% accuracy. But when talking about all Devanagari character it shows mistake in recognising some character. Recognition rate of all Devanagari character is near about 92.66%.

REFERENCES

1. Madhu Shahi, Dr. Anil K Ahlawat, and Mr. B.N Pandey, *"Literature Survey on Offline Recognition of Handwritten Devnagari Curve Script Using ANN Approach."* International Journal of Scientific and Research Publications, Volume 2, Issue 5, May 2012 1 ISSN 2250-3153

2. R. Jayadevan, Satish R. Kolhe, Pradeep M. Patil, and Umapada Pal *"Offline Recognition of Devanagari Script: A Survey"*, ieee transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 41, no. 6, november 2011.

3. U. Pal and B. B. Chaudhuri, "*Indian script character recognition: A survey*," Pattern Recognit., vol. 37, pp. 1887–1899, 2004.

4. B. B Chaudhuri and U. Pal, "*An OCR system to read two Indian language scripts: Bangla and Devanagari*," in Proc. 4th Conf. Document Anal. Recognit., 1997, pp. 1011–1015.

5. V. K. Govindan and A. P. Shivaprasad, "*Character recognition: A survey*," Pattern Recognit., vol. 23, pp. 671–683, 1990.

6. JanFlusser and Tomas Suk, "*Pattern Recognition by Affine moment invariants*." Pattern Recognition, Vol.26,No.1,pp.167,174.1993 Printed in great Britain.

7. Kailash S. Sharma,A. R. Karwankar, Dr. A.S.Bhalchandra," *Devanagari Character Recognition Using Self Organizing Maps*" ICCCCT'10

8. N. Sharma, U. Pal, F. Kimura, and S. Pal, "*Recognition of offline handwritten Devanagari characters using quadratic classifier*," in Proc. Indian Conf. Comput. Vis. Graph. Image Process., 2006, pp. 805–816.

9. U. Pal, T. Wakabayashi, N. Sharma, and F. Kimura, "*Handwritten numeral recognition of six popular Indian scripts*," in Proc. 9th Conf. Document Anal. Recognit., 2007, pp. 749–753.

10. R.M.K. Sinha, and Veena Bansal*, "On Automating trainer for construction of prototypes for Devanagari text recognition"* Technical report TRCS-95-232, IIT Kanpur, India 1995

11. U. Pal, P. K. Kundu, and B. B. Chaudhuri*, "OCR error correction of an Inflectional Indian language using morphological parsing," J. Inf. Sci. Eng.*, vol. 16, no. 6, pp. 903–922, 2000

12. U. Pal and B. B. Chaudhuri, *"Printed Devnagari script OCR system,"* Vivek, vol. 10, pp. 12–24, 1997.

13. B. B. Chaudhuri and U. Pal, "*Skew angle detection of digitized Indian script documents,"* IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 2, pp. 182–186, Feb. 1997.

14. S. Kompalli, S. Setlur, and V. Govindaraju, "Devanagari OCR using a recognition driven segmentation framework and stochastic language models," Int. J. Document Anal. Recognit., vol. 12, pp. 123–138, 2009.

15. H.Ma and D. Doermann, "Adaptive Devnagari OCR using generalized Hausdorff image comparison," ACM Trans. Asian Lang. Inf. Process., vol. 2, no. 3, pp. 193–218, 2003.